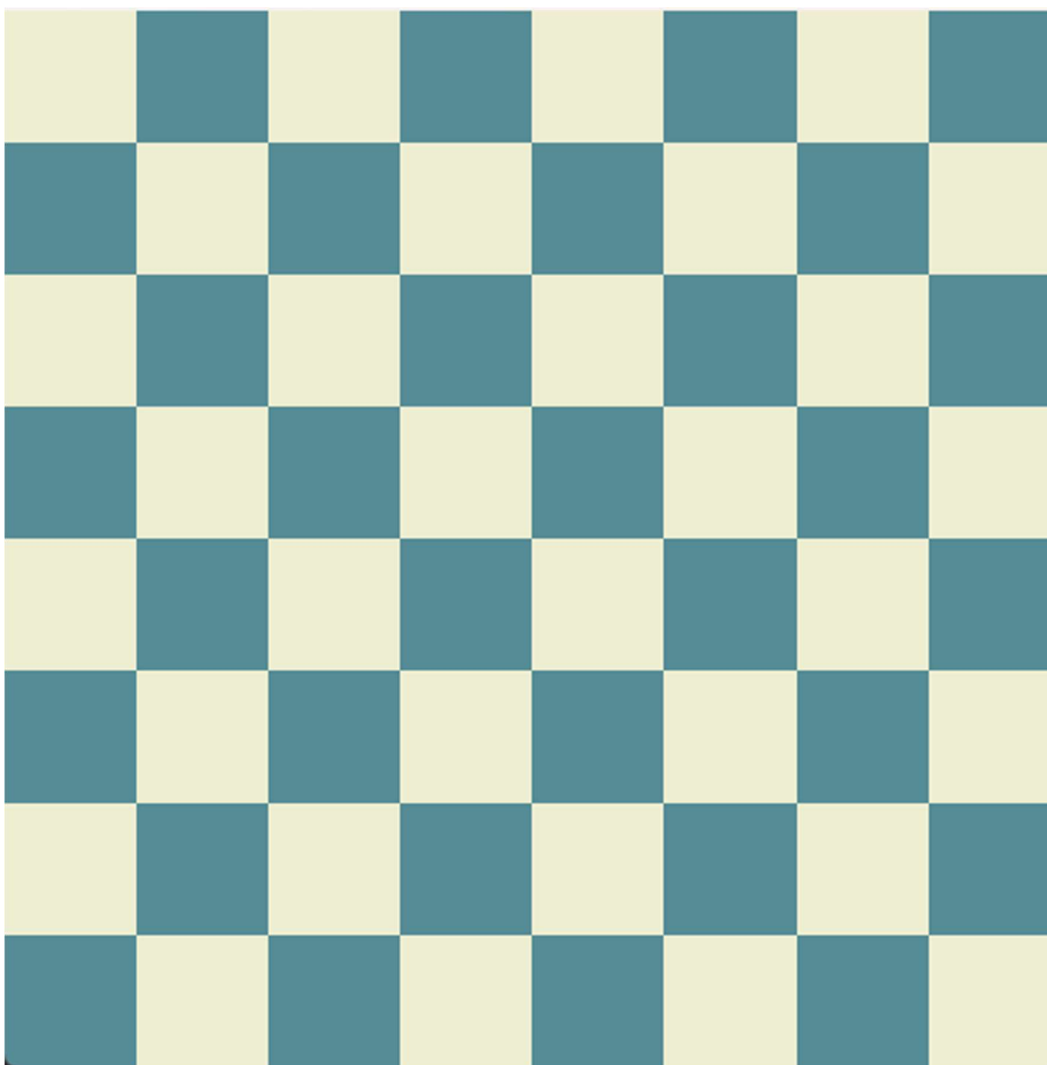


Počáteční problém

Tématem semestrální práce byla jednoduchá vizualizace šachovnice, společně se šachovými figurkami, které na ní budou přesně podle pravidel rozmístěné. Dále mělo být možné s figurkami pohybovat dle pravidel standartní hry a tento pohyb měl být animován. Šachy také měli obsahovat různá pravidla, která jsou standartně používána. Je to braní mimochodem, rošáda a proměna. Pokud při hře dojde k matu či patové situaci, hra bude ukončena.

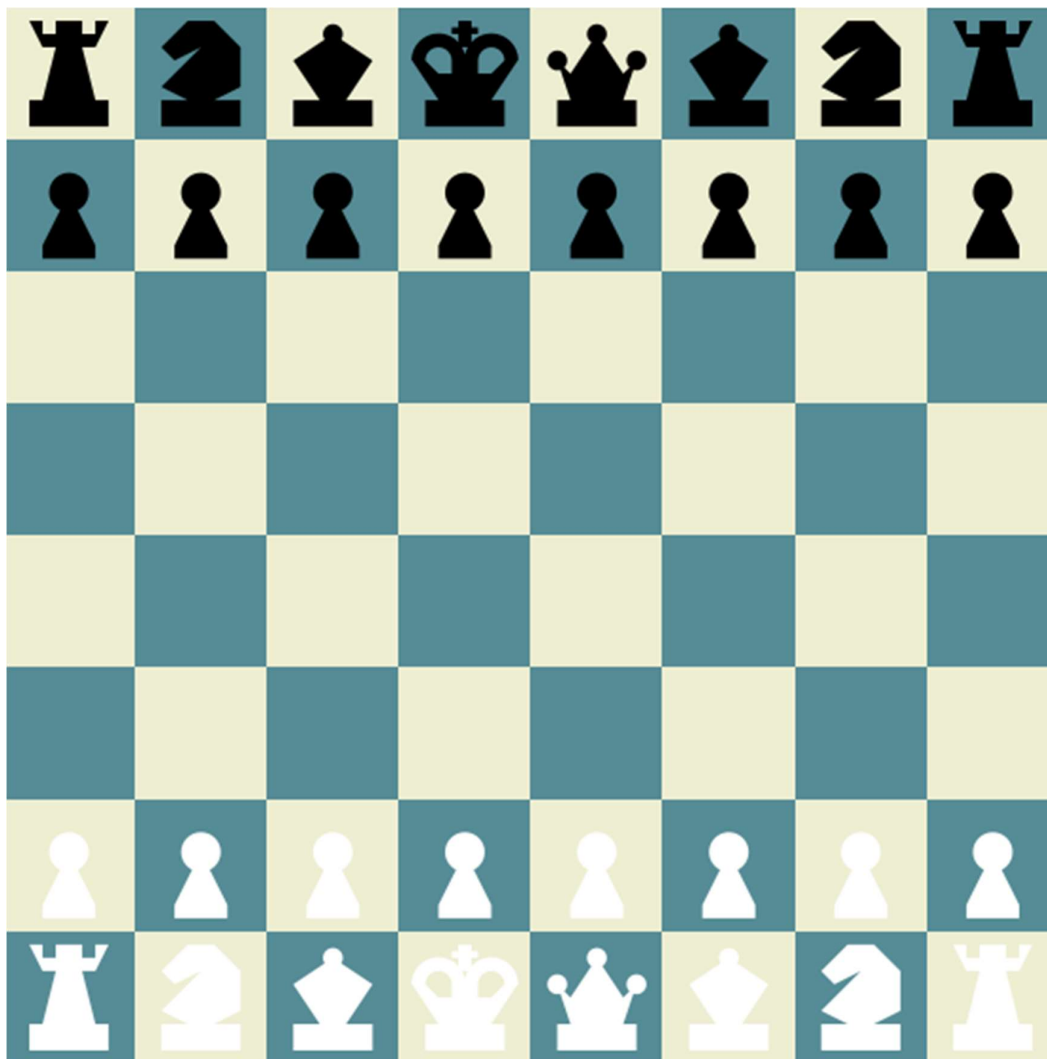
Řešení problému



Obr. 1 – vizualizace šachovnice

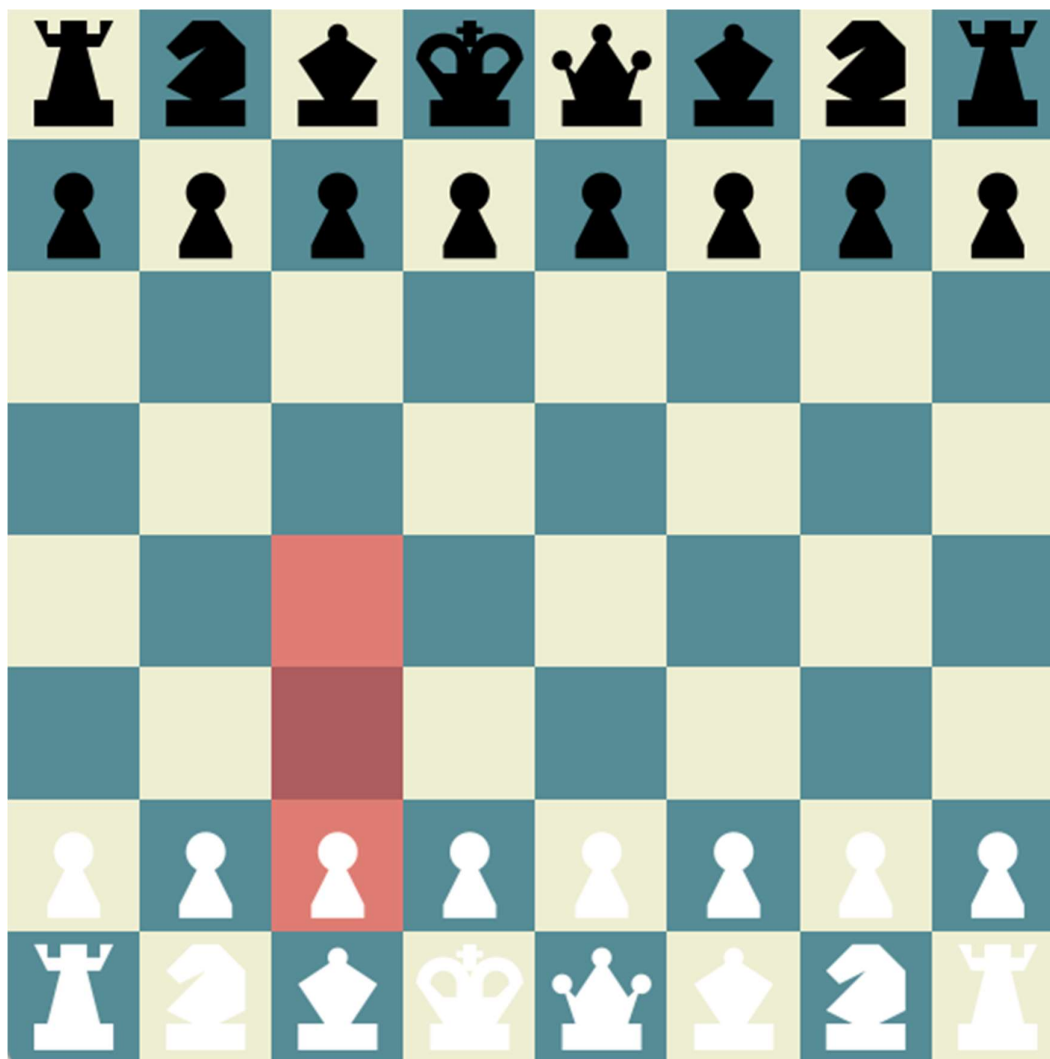
Nejdříve bylo třeba vizualizovat podklad, šachovnici (viz. Obr. 1). Postupoval jsem tak, že jsem nejprve vykreslil 8 řádek po 8 čtverečkách, které ve výsledku představují jednotlivá políčka šachovnice. Poté bylo třeba vykreslit všechna políčka se správnou barvou. Toho jsem docílil tak, že si vždy zjišťuji, zda vykresluji čtvereček na liché či sudé pozici na dané řádce a následně, zda je aktuální řádka lichá či sudá. Podle těchto informací se poté nastaví barva, kterou bude výsledné políčko vyplněno. Jedním z kritérií také bylo, aby se šachovnice vždy držela ve středu okna, v závislosti na tom, jak uživatel upraví jeho velikost. Tento, zprvu poměrně náročný úkol, jsem nakonec vyřešil tak, že za běhu programu zjišťuji, zda není větší šířka než výška

či naopak, nebo, zda jsou obě hodnoty stejné. Následně, podle příslušných pokynů, nastavuji, o kolik se, v prvním případě, má šachovnice posunout zleva, nebo, v druhém případě, od shora okna, ve kterém je šachovnice vykreslena.



Obr. 2 – vizualizace šachových figurek

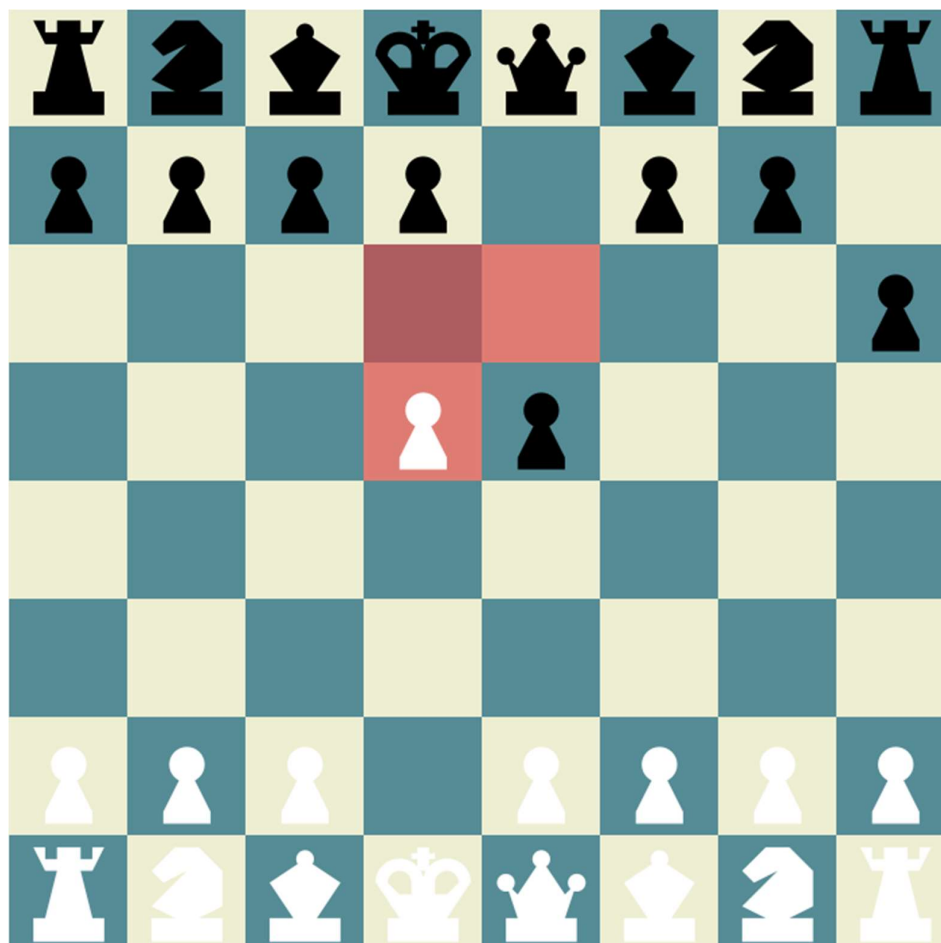
Dalším úkolem bylo navrhnout a vykreslit šachové figurky (viz. Obr. 2). Všechny figurky jsou vykreslovány stejným principem. Funguje to tak, že si políčko rozdělím vertikálně i horizontálně na 10 stejných dílků a pomocí souřadného systému následně jednoduše zjistím body, které po spojení vytvoří figurku a ta bude vypadat přesně tak, jak jsem zamýšlel. Opět, stejně jako u šachovnice, se figurky musí měnit v závislosti na velikosti okna. Díky tomu, že používám výše zmíněnou metodu, se mi velikost figurek automaticky upravuje a já, aniž bych to zprvu tušil, implementoval řešení již na začátku. Poté už bylo jen třeba umístit figurky na svá místa a držet je vždy ve středu políčka. Řeším to tak, že mám uložené souřadnice jednotlivých políček a pak pouze nastavím, na jakém se má příslušná figurka vykreslit a opět se o všechno ostatní postará souřadný systém.



Obr. 3 – vizualizace volných tahů figurek

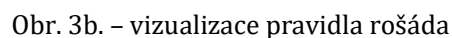
Pohyb figurek je konstruován tak, že se nejprve zjistí, jaká figurka je vybrána a jaký hráč si jí vybral. Následně, po stisknutí na figurku, se na šachovnici zobrazí červená políčka (viz. Obr. 3), která reprezentují potenciální místa pohybu figurky. Program zvaliduje tah v souladu s pravidly šachu a pokud je tah validní a hráč si je jistý svým tahem, stačí přetáhnout myš na určité, zvýrazněné, políčko a pustit. Funkce přesunu je tedy následující. Klikne se na příslušnou figurku, levé myšítko se podrží do doby, kam se má figurka přesunout a na tomto místě stačí myš pustit a figurka se, za doprovodu animace přesune na zvolené políčko. Na řadě je pak hráč druhý, který opakuje stejný postup.

V programu je také obsažena trojice pravidel, která jsou obsažena i v klasických šachách.



Obr. 3a – vizualizace pravidla braní mimochodem

Jako první je zde ošetřeno braní mimochodem (viz. Obr. 3a.). Tato funkce nastává pouze tehdy, kdy hráčův pěšák je hned vedle pěšáka, který se poprvé posunul o dva tahy. Pokud se tomu tak stane, nad příslušným pěšákem protivníka se zobrazí přístupný tah hráčova pěšce a hráč je schopen tímto úkonem vyhodit sousedního protihráčova pěšce.



Pokud je volena rošáda doprava, král se přesune na nejkrainější pozici vpravo. Současne s posunem krále se začne posouvat i věž, která se zastaví na první pozici vlevo od krále, tudíž král bude mít věž po své levici. Pokud je volena rošáda doleva, král se přesune na nejkrainější pozici vlevo. Současne s posunem krále se začne posouvat i věž, která se zastaví na první pozici vpravo od krále, tudíž, v tomto případě, bude mít král věž po své pravici.



Obr. 3c – reakce na proměnu figurky

Třetím a posledním pravidlem je proměna (viz. Obr. 3c.). Pokud se jeden z pěšáků dostane na konec hracího pole, tudíž pozice bude 1 nebo 8, zobrazí se okno se čtyřmi tlačítky a je zde možnost vybrat si, za co bude pěšák proměněn. Je zde zakomponována kompletní proměna, tzn. že má hráč na výběr, zda pěšáka proměnit na koně, střelce, věž nebo královnu. Po stisknutí tlačítka se původní pěšák smaže a nahradí se novou figurkou dle výběru hráče.

Pokud dojde k matu či patové situaci, hra vhodně zareaguje a zobrazí výsledek hry.



Obr. 4 – reakce na konec hry

Funkce matu spočívá v tom, že se v každém kroku kontroluje, zda protihráč má, po tahu hráče, nějaký možný tah. Je zde metoda, která kontroluje pohyby všech figurek a vyhodnocuje, zda je tedy schopen protihráč ještě hrát či nikoliv a v této chvíli by přišel o krále. Je zde také kontrola, zda soupeřův král není v šachu.

Pokud nastane moment, kdy protihráč už nemá žádný volný tah, hra zahlásí, že hra byla zakončena matem a jaký hráč hru vyhrál (viz. Obr. 4). Následně si hráč může zvolit, zda hru restartovat či úplně odejít.

Na obrázku můžeme vidět, že se černý snaží o šach krále. Zjistí se, zda po přesunu figurky má protihráč nějaké volné tahy a zda je možné nějak zabránit šachu. Pokud se u soupeře nenachází žádný volný tah, který by zabránil vyhození krále, hra skončí a zobrazí se, který hráč hru vyhrál.

Obdobnou funkcionalitu najdeme u kontroly patové situace. Tam se opět kontrolují možné tahy soupeře, ale je podmínkou, že král nesmí být v šachu, aby bylo zakončení patem. Pokud tedy k takové situaci dojde, na obrazovce se objeví stejné okno, jako v předchozím případě, jen se nebude jednat o ukončení hry matem, ale patem. Opět jsou zde možnosti si zvolit restart či ukončení hry.